

Computers and Internet  
CQ Magazine, September 2002  
Don Rotolo, N2IRZ

"Plastics. There's a great future in plastics." Back in 1967, that was good advice. Here in 2002, I'll give you some more good advice: "Software. The world runs on software". Remember that one.

The great software boom started, in my own opinion, around the late 1980's, when video games and computers were available to anyone for a reasonable price. Some decades later, software is still one of the few products you can make and sell which costs virtually nothing to reproduce, warehouse or distribute.

This month, I'd like to explore Programming a little bit. Programming, as in writing software to make microprocessors and computers do things that you find useful. Now, before we get started, please understand that there is no way that I can teach you how to program *anything* in the space allocated to this column. A good tutorial wouldn't fit in a year's worth of CQ. But, that doesn't mean you can't learn something today, and maybe gain a toehold into something sure to enhance your enjoyment of Amateur Radio.

Programming can be as simple as storing a frequency in your HT's memory, or as complex as a full-blown application written in Visual Basic. We have hundreds of opportunities to program things. While this can lead to confusion, the proliferation of different ways of writing computer software (called "code" by programmers) has actually made it much easier to be learned by regular folks like you and me. With such a wide variety available, there's something that'll fit your needs and skills perfectly.

## Languages

Before I go further, let's talk about programming languages. Just like in human speech, there are many programming languages available to us. You can more or less accomplish the same things with most any programming language, although some are better suited to certain tasks than others. These languages have names like BASIC, COBOL, Pascal, C++, Fortran, Java, and the like.

Older languages, which were originally intended to be run on large mainframe computers, tend to be Linear in nature. That is, each instruction is on it's own numbered line, and the computer executes each line in order. Newer languages, written for the Graphical User Interface (GUI) environment (such as Windows) tend to be Event-driven, where some event (such as a mouse click) causes a little snippet of the program to execute.

BASIC (an acronym for Beginners All-purpose Symbolic Instruction Code) is a great language for a beginner to learn. Although it is a linear language, it exposes you to all the major programming concepts and types of commands available in most languages, is reasonably powerful, and readily available. BASIC in its regular form is included with most versions of DOS (I said it was Older!), and has been adapted to many simpler microprocessor platforms, the most notable of which is the Basic Stamp from Parallax Corporation. I strongly recommend that you try some form of BASIC for a first effort.

For more advanced programming efforts, look towards a more modern Object-Oriented Programming (OOP) language, such as C++ or any of the Microsoft 'Visual' series. My current expertise is in Microsoft Visual Basic (which, despite the name, is very unlike BASIC), which combines OOP with simplified GUI programming.

## OOPs

For the record, OOP is a very powerful concept in modern programming, but often misunderstood. OOP is a concept where you bundle more complex events or processes into an 'object'. It becomes a lot easier to handle the 'object' than it's individual pieces. This greatly simplifies operations which are very complex (but not impossible!) in non-OOP languages.

For example, you could write a program that treats a Microsoft Word document as an object. That makes any operations on the document very simple, because you don't care about what is inside the object. To print a document from Visual basic using the default printer, you would simply write **Document1.Print**, and it would happen. You can also change the object's properties, for example **Document1.Orientation = Landscape** and when you send it to the printer it will print sideways. I hope this helps you understand that an object is just a convenient way of bundling something up so it's easier to program with, and it's not a concept to be afraid of.

Enough of that, let's look at some of the things you can do:

## Robots

I bought a Lego MindStorms robot kit for my 7 year old daughter the other day. I had promised her we'd build a robot sometime, and I had been considering one I saw in the Jameco catalog, which follows a line drawn on the ground. She, on the other hand, wanted a robot that would clean her room for her. After I explained the slight differences between the two projects, she agreed that maybe we should start simple and work our way up.

The MindStorms programming language is simple to learn, and the hardware couldn't be easier (They're Legos!). You build a robot, using motors and sensors, write a program on

your regular computer and load it into the controller module, where it (hopefully) does what you wanted it to do. While the programming language that comes with the MindStorms kit is somewhat limited, there are others available (on the Internet) that allow you to do more.

## Computers

Some years ago, I had a business problem that wouldn't go away. It involved cataloging the locations of various parts of a complex assembly, so a repair technician could locate the parts he was looking for. I started keeping a notebook of where these parts were, but that became tedious to maintain., Instead, I got a copy of Visual Basic (VB) and wrote a kind of database program which showed where the parts were with a photo and descriptive text. The program was a hit, and ended up being distributed worldwide.

Before I started, I'd never used VB. I was fooled by the name into thinking it was just like the BASIC I used to use in high school. I ended up buying a book and learning through trial and error (mostly error) how to write VB Code. The point is, don't try to learn how to program until you have something in mind that you want to accomplish. That way, you have a goal and can learn as you go. Without a goal, it can become an exercise in frustration.

As an added bonus, nearly every Microsoft application today (such as Word, Excel, Access, etc) uses a version of VB for it's 'macro' language, known as Visual Basic for Applications (VBA). If you know VB, you know most of VBA, and vice-versa. Think of how valuable that skill could make you at work. I know what it's done for me!

## Microprocessors

Some years ago, I helped write a paper for the 15<sup>th</sup> Digital Communications Conference on a 9600 baud G3RUH-type Packet modem for the parallel port, implemented using a PIC microprocessor. The computer did most of the TNC work, but the modem had to provide the link between data and audio. The hardware was surprisingly simple, the whole trick was in using some fancy software. Eventually marketed as the PICpar modem, it allowed users to implement 9600 baud packet stations for just a few dollars.

The Basic Stamp series by Parallax, and many similar products, are designed for ease of programming. Powerful enough to do most any smaller job, inexpensive enough to lose, and easy enough for a grade-schooler. The reference materials alone are worth studying, and you can make some really nifty and useful projects from almost thin air. Morse code keyers or decoders, a balloon controller, timers, robots - the possibilities are endless. This is an excellent way to get into programming, and you can create useful and really marketable products for a very small investment.

## The Rest

Most anything electronic has some software inside it. While the software in your new HT is probably hard-coded in silicon (which is less expensive and more durable than using programmable memory), your large HF rig might actually be reprogrammable. At least the DSP filters, which are pure software, are fair game. DSP has other uses besides filtering, including modulation, demodulation, and even spectrum displays.

How do you think all those logging programs got started? Yep, a ham with a problem to solve. How about a better rotator controller? I haven't even begun to talk about internet programming, what with Active-X, Java, and even HTML. Ran out of space already!

I guess what I'm trying to say is this: Software is everywhere, the possibilities are literally endless. It isn't difficult or expensive to write your own. Try it, you might be surprised.

Until next time,  
73, Don N2IRZ

## Resources:

BASIC Language: While there are many on-line resources, try <http://dmoz.org/Computers/Programming/Languages/BASIC/> for some excellent links.

LEGO makes the MindStorms line of programmable robots and accessories. Visit them at <http://mindstorms.lego.com/>

Microsoft offers a wide range of programming languages, including their popular and powerful Visual series. Visit them at <http://www.microsoft.com/catalog/navigation.asp?subid=22&nv=3>.

Parallax Inc. makes the Basic Stamp series of microcontrollers. Visit them at <http://www.parallaxinc.com/> and download free documentation and utilities. Also see <http://www.basicmicro.com/> for a competitor's similar products and <http://www.nutsvolts.com/stmpindx.htm> for some great articles about Stamps.

PICs are manufactured by Microchip, Inc. Visit them at <http://www.microchip.com/index.asp>.

At the Tucson Amateur Packet Radio site <http://www.tapr.org/tapr/html/Fcnc15.html> you can listen to a presentation about the PICpar modem, given by Keith Sproul, WU2Z when I couldn't attend at the last minute.

The 'Plastics' quote comes from the 1967 film "The Graduate". Visit <http://www.filmsite.org/grad.html>, or browse to <http://www.sdplastics.com/oneword.ram> for a sound bite.

Sidebar:

In just a few days, TAPR and the ARRL will present the 21<sup>st</sup> annual Digital Communications Conference. This year, it's being held in Denver. If you are within driving distance, or can somehow get to Denver for the weekend, you will not be disappointed. This is the annual meeting of the do-ers in digital communications, a treat not to be missed. For more info, visit <<http://www.tapr.org/tapr/dcc/>>